Automated Machinery Health Monitoring
Using
Stress Wave Analysis & Artificial Intelligence

David B. Board
DME Corporation
6830 N.W. 16th Terrace
Fort Lauderdale, FL 33309
(954) 975-2100

1998

Abstract: This paper describes the current state of development of a prototype mechanical diagnostic system being developed for the U.S. Army, for application to helicopter drive train components. The system will detect structure borne, high frequency acoustic data, and process it with feature extraction and polynomial network artificial intelligence software. Data for network training and evaluation has been acquired from both healthy and discrepant components, operated over a full range of loads, in a test cell.

Stress Wave Analysis (SWAN™) is a high frequency acoustic sensing and signal conditioning technology, which provides an analog signal that is a time history of friction and shock events in a machine. This "Stress Wave Pulse Train" (SWPT) is independent of background levels of vibration and audible noise. The SWPT is digitized and used to compute a set features that characterize the "friction signature". Fault Detection Networks of polynomial equations are used to automatically classify SWPT features as being representative of either healthy or discrepant mechanical components. The application of these techniques for automatic classification of friction signatures advances current technology to achieve real time diagnostic capability ***at all flight power levels***.

Keywords: Health Monitoring; Stress Wave Analysis (SWAN); Artificial Intelligence; Predictive Maintenance; Poylnomial Network Modeling; Structure-borne Ultrasound;

System Description: The Distributed Stress Wave Analysis (DSWAN) system consists of stress wave sensors, interconnect cables, and three types of modules: Distributed Processing Units (DPU's), a Maintenance Advisory Panel (MAP), and a LapTop Computer (LTC). The sensors, DPUs, and MAP are airborne components of the system. The LTC is ground based, and is not required for in-flight autonomous operation of the DSWAN airborne components.

Each DPU scans up to 8 sensor locations, extracts the friction and shock signal from broadband noise, and uses Fault Detection Network (FDN) software to detect abnormal friction/shock signatures from the monitored components. The monitored components

include all the H-60 primary drive train elements except the engines. Although up to four DPU's can be connected to one MAP, only two will be required to monitor the 15 H-60 drive train sensors. The DPU also contains Diagnostic & Prognostic Network (DPN) software for Sensor Validation Networks (SVN's), Regime Recognition Networks (RRN's), and self test.

When a DPU detects a potential problem, it turns on an associated indicator on the MAP. The LTC can then be used, during a post flight inspection, to download, analyze, and display, detected and forecasted problems. The LTC can also be used to upload new software to the DPU memory, or to reprogram a DPU for use with a different set of sensors.

Data Base Description: The data employed in the development and evaluation of the various types of DPN's was acquired from the U. S. Navy's H-60 test cell, located at the Naval Air warfare Center in Trenton, NJ. This facility consists of an entire shipset of H-60 helicopter drive train components, (including turboshaft engines) with associated loading mechanisms, to exercise the drive train over a full range of operating loads. Over a period of 32 months, from August of 1994 through April of 1997, baseline and seeded fault drive train components were tested. High frequency acoustic (stress wave) data was acquired from 15 sensor locations (listed in Table I) using a DME SWAN 3000 data acquisition system and a TEAC Digital AudioTape (DAT) recorder. The PC based SWAN 3000 scanned the 15 sensor locations, then filtered and demodulated the high frequency acoustic data to provide a demodulated Stress Wave Pulse Train (SWPT) for recording on the DAT.

Case Descriptions: The following paragraphs describe the baseline and seeded fault ("CASE") test configurations used in training and evaluating DPN's from the main transmission assembly data base.

<u>Baseline MM02:</u> This was one Main Module with non discrepant components and Input Modules, but it was tested first in August of 1994, then used for a series of seeded fault tests, before being rebuilt with all good parts and retested in August of 1995. Thus we have data from two "builds" of a main transmission assembly. Data from the 1994 runs is available only at one load condition (TQMR = 38K) and the 1995 runs have no data from sensor 3. By combining the test data from both builds/runs we have a comprehensive database for all sensors at all loads.

<u>Baseline MM0396:</u> This was a completely different build of Main Module 03, and two Input Modules, tested in February of 1996. There were no known discrepant parts in any of these three main transmission assembly modules, but a shim survey was being conducted, and no shim survey was completed.

<u>Baseline MM0397</u>**:** This was a completely different build of Main Module 03 and two Input Modules tested in May of 1997. There were no known discrepant parts in any of these three main transmission assembly modules, and no shim survey was conducted.

Table I: H-60 Drive Train Stress wave Sensor Locations

Stress Wave Sensor Number	Sensor Location
1	Port Input Module Input Housing
2	Port Input Module Output Housing
3	Starboard Input Module Input Housing
4	Starboard Input Module Output Housing
5	Main Module Planetary Ring Gear
6	Main Module Upper Cover
7	Tail Rotor Output/Rotor Brake Support Bracket
8	Forward Tail Rotor Shaft Bearing
9	Mid Tail rotor Shaft Support Bearing
10	Forward Disconnect Coupling Support Bearing
11	Intermediate Gearbox Disconnect Coupling Support Bearing
12	Intermediate Gearbox Input
13	Intermediate Gearbox Output
14	Tail Rotor Gearbox Input
15	Tail Rotor Gearbox Output

Case 1: Spalled rolling element (spherical roller) in a planetary gear. This was a fleet removal for debris generation.

Case 2: Main transmission starboard Timken (tapered roller) bearing. This was a fleet removal for debris generation.

Case 3: a) Main Module port input pinion gear spall (fleet reject).
b) 1/3 of a tooth removed from an Input Module input pinion. The pinion was machined to remove 1/3 of the working tooth at the toe side.

Case 4: Integral race spall in the Main Module starboard input pinion.

Case 6: Main Module port input pinion gear spall (fleet reject).

Case 7: Electron Discharge Machined (EDM'd) roller bearing in the Starboard Input Module, and EDM'd ball bearing in the Port Input Module.

Case 8: Electron Discharge Machined (EDM'd) roller bearing in the Starboard Input Module; EDM'd ball bearing in the Port Input Module; and Main Module Planet Gear Tooth fault.

Case 9: Same as CASE 7 (Electron Discharge Machined (EDM'd) roller bearing in the Starboard Input Module, and EDM'd ball bearing in the Port Input Module) with the addition of high vibration of High-Speed shaft.

MM02 baseline data and seeded fault cases 1, 2, 3, 4, and 6 were used to train and evaluate FDN's. MM03 and case 7, 8, and 9 data have not yet been added to the training database or used for evaluation of the FDN's developed thus far.

Data Reduction & Feature Extraction

Data Reduction: Approximately 160 hours of Stress Wave Pulse Train (SWPT) data were recorded during the 32 months of testing. This data is contained on 79 tapes that were correlated and cataloged with the Microsoft Access database, from the SWAN 3000 data acquisition system, and the Test Cell Data Logs provided by NAWC Trenton.

Data reduction began with the laborious task of locating specific data records on each of the tapes. The SWAN 3000 is a 2-channel data acquisition system. It was set up to lock channel 2 onto one sensor for continuous data recording during a test, while channel 1 would sequentially scan the remaining 14 sensors, plus a signal tone. The analyst first used the Access data base and the test cell log sheets to locate a particular operating (load) condition on the tape for a given test date. Then the tape was monitored to listen for the signal tone that identifies the end of one scanning sequence and the beginning of another. Data records were 15 seconds in duration, so by using an oscilloscope and a stopwatch it was possible to identify which of the sensors was the source of the recorded SWPT at any point on the tape. Each sensor's elapsed time location on tape was then entered into the data reduction log, so that it could be more easily retrieved for subsequent digitization, feature extraction, network synthesis, and system test.

The next step was to digitize the data from each sensor, at each load, for each test configuration. This was accomplished using the Transient Capture (TC) feature of the SWAN 3000. The analog SWPT data has a frequency content of 0 to 5000 Hz, before attenuation due to roll off characteristics of the demodulator in the system's analog signal conditioner. The SWPT data is therefore digitized at a 20 KHz sample rate (with 12 bit resolution) to satisfy Nyquist anti-aliasing criteria. Typically the "middle"10 -12 seconds of each record were used to create 5 Time History (TH) files (each of 2-second duration). This avoided including switching transients in the data, and provided files of the same length planned for use in the DPU. To date, approximately 2,200 of these TH files have been generated and stored in a computer directory as binary files.

Feature Extraction: Specialized Feature Extraction (FE) software has been developed for the purpose of accurately characterizing the SWPT and compressing the TH files. This custom SWAN Feature Extraction software is unique to the interpretation of the Stress Wave Pulse Train (SWPT) for the quantitative analysis of friction and shock events in operating machinery. The FE software is modular, and is available in two versions: the Analyst Mode, and the DPU Mode. The Analyst Mode includes a " WINDOWS shell" written in Visual Basic and is used in the PC environment to develop input tables for the

synthesis and test of DPN's. The DPU Mode is the operational form of the code, as required by the DPU and LTC components of the DSWAN system. Feature Extraction starts with the TH file of the SWPT. Mathematical transforms are then applied to the time series data for characterization of 37 time domain waveform features such as pulse amplitude, duration and energy content.

Fault Detection Networks (FDN's): FDN's are resident in each of the DPU's. Sufficient non-volatile memory has been allocated to store up to 32 FDN's in each DPU. These 32 FDN's will perform the function of "anomaly detection" for the 8 sensors connected to the DPU. When an anomalous condition is detected, the features will be stored for postflight download to the LTC ground station. The LTC software will then combine this data with trended data, and data from any other related sensors, for Enhanced Fault Detection, Fault Location, Fault Isolation, and estimating Percent Degradation.

The inputs to the FDN's are the time domain features extracted from each 2-second Time History file at each sensor location. The Analyst Mode of the TDFE software was used to extract the time domain features that were then used to train and evaluate the FDN's for sensor locations 1 through 5 on the H-60 main transmission assembly. MM02 baseline data and seeded fault cases 1, 2, 3, 4, and 6 were used to train and evaluate FDN's.

Before FDN synthesis can begin, input data tables of examples must be carefully constructed to train and evaluate various possible network configurations. Each row in one of these input tables is an example of Stress Wave Pulse Train time domain features for a given test configuration (baseline or seeded fault) and operating condition (main rotor torque (TQMR)). Each column, except the last, in an input table corresponds to one of the 37 time domain features. The value in the last column identifies the features in that row as either a baseline (0) or seeded fault (1) example.

A table must be prepared for each sensor location, and a separate set of five tables must be prepared for each of the five-seeded fault conditions. Each of these tables contains approximately 100 to 200 examples of both seeded fault and baseline conditions. About 1/3 to 1/2 of the data are from seeded fault conditions, and represents a full range of operating loads.

The basic strategy employed in the synthesis of the FDN's was as follows:

1. Train and evaluate first on a full normal load range of data. Train FDN's based upon two or more load ranges only if necessary to achieve satisfactory diagnostic accuracy.

2. Use a set of seeded faults that represent a range of fault types (gears and bearings) fault locations (input modules and planetary gear system) and damage levels.

3. First synthesize FDN's that are optimized for the detection of a specific fault type and location (fault specific). Then, given satisfactory accuracy for fault specific models, synthesize networks that are more generic, and capable of accurately diagnosing multiple types of faults.

4. Train all test models using 75% of the data, and reserve 25% of the data for FDN evaluation.

5. Employ parametric constraints in the modeling/synthesis process, to limit the complexity of the resulting FDN's. This is necessary both to limit the amount of executable code required to implement a model, and to avoid "over-fitting" the network to the available database.

Optimizing the Complexity Penalty Multiplier (CPM): The software used to synthesize and evaluate the network of polynomial equations (AbTech's ModelQuest Expert) has a user setable parameter called the Complexity Penalty Multiplier. The CPM is used to minimize a modeling criterion that attempts to select as accurate a network model as possible, without over fitting the data. Over fitting occurs when the network model becomes so specific to the training data that it does a poor job of evaluating new data. ModelQuest minimizes over fit by performing a trade-off between model complexity and accuracy, based on the assumption that simpler models are more general and superior for as yet unseen data (i.e. data not used for training).

In order to optimize the CPM for a given network, a model is set up with a given input table and set of modeling parameters such as the maximum number of hidden layers and the maximum number of inputs. This model is then exercised through approximately 10 iterations of varying the CPM and assessing the impact on the average absolute error of the network output. For the H-60 main transmission assembly, this iterative process was repeated for each of the five sensor locations for each of the five seeded fault cases. Then the FDN was resynthesized at the best CPM for each of the five sensor locations and each of the five seeded faults. (Thus the CPM optimization process required 275 modeling iterations.)

This entire CPM optimization process was repeated twice: first using all the data to train and 25% of the data (which had been used in training) to evaluate; then using 75% of the data to train and 25% to evaluate. The purpose of this exercise was to assure that the CPM was optimized based upon the full range of data, and to check if there was a significant difference in the optimized CPM when trained on a reduced data set. (If the optimum CPM from the reduced data set was significantly different, it would indicate the need to change the Random Number Seed in the model setup so that the full **range** of data would be used when training on only 75% of the data base.) In all cases, the FDN's saved for evaluation purposes, were those that were trained on 75% and evaluated on 25% that was not used in training.

Optimizing the Evaluation Threshold: The next step was to optimize the evaluation threshold for the "best CPM" model at each sensor location for each of the fault cases.

(This threshold is the value of the network output, between 0 and 1, above which a fault message is generated.) This optimization process consists of making a tradeoff between False Alarms and False Dismissals (undetected faults).

In the case of an aircraft like the H-60, the consequences of a false alarm are worse than that of a false dismissal. This is due to two principal reasons:

1. The probable consequences of a false inflight alarm range from a mission abort to an accident resulting from the hazards associated with a precautionary landing. A false post flight alarm will result in an unnecessary maintenance action, and possibly a premature, or incorrect, component removal.

2. The probable consequences of an undetected fault are not as serious, mainly because there will be more opportunities to detect the problem as additional measurements are made. With each additional scan of the sensor and application of the FDN, the cumulative probability of detection will increase. Since the sensor will be scanned at least once every 18-20 seconds, there will be about 200 opportunities for the FDN to find the problem during each hour of operation. As the problem develops during subsequent hours of operation, the symptoms of the problem will also become increasingly clear, and the probability of detection will be increased even further.

For these reasons, and prior quantitative analysis of their implications for helicopter fleets, the alarm threshold must almost always be optimized to cut the false alarm rate to a minimum. 25 iterations of threshold optimization were performed: one for each of 5 sensor locations, for each of the 5 seeded fault conditions. Tables II through VI show the false alarm and false dismissal rates for the best of these **fault specific** FDN's at each of five sensor locations, for seeded fault cases 1, 2, 3, 4, and 6. These results show that for each seeded fault, at least 3 out of five sensor locations had FDN's that were 100% accurate, when tested on data that had not been used in training. This was sufficiently encouraging to proceed to the next step of the overall FDN development strategy: synthesis and test of FDN's capable of detecting multiple types of seeded faults.

Table II: FDN Diagnostic Accuracy, Case 1
(spalled planet bearing roller)

Sensor Location	Evaluation Threshold	False Alarm %	False Dismissal %	Comments
1	0.50	0.00	0.00	
2	0.50	0.00	0.00	
3	0.65	5.70	5.70	
4	0.65	2.80	2.80	
5	0.50	0.00	0.00	closest sensor & most accurate FDN

Table III: FDN Diagnostic Accuracy, Case 2
(MM input pinion Timken Bearing)

Sensor Location	Evaluation Threshold	False Alarm %	False Dismissal %	Comments
1	0.50	0.00	0.00	
2	0.50	0.00	0.00	closest sensor
3	0.40	0.00	0.00	
4	0.50	0.00	0.00	
5	0.50	0.00	0.00	most accurate FDN

Table IV: FDN Diagnostic Accuracy, Case 3
(MM input pinion gear spall, IM input pinion ground-off tooth)

Sensor Location	Evaluation Threshold	False Alarm %	False Dismissal %	Comments
1	0.55	0.00	0.00	
2	0.66	3.10	3.10	closest to gear spall
3	0.50	0.00	0.00	
4	0.65	0.00	0.00	
5	0.50	0.00	0.00	most accurate FDN

Table V: FDN Diagnostic Accuracy, Case 4
(MM input pinion roller bearing spall)

Sensor Location	Evaluation Threshold	False Alarm %	False Dismissal %	Comments
1	0.50	0.00	0.00	most accurate FDN
2	0.45	0.00	0.00	
3	0.50	0.00	0.00	
4	0.50	0.00	0.00	closest sensor
5	0.50	0.00	0.00	

Table VI: FDN Diagnostic Accuracy, Case 6
(MM input pinion gear spall)

Sensor Location	Evaluation Threshold	False Alarm %	False Dismissal %	Comments
1	0.65	0.00	5.70	
2	0.50	0.00	0.00	closest sensor & most a accurate FDN
3	0.50	0.00	0.00	
4	0.50	0.00	0.00	
5	0.45	0.00	2.90	

Three approaches were considered for development of multi-case FDN's:

1. Develop two FDN's; one for gear faults, and the other for bearing problems, at each of the five sensor locations.

2. Develop one FDN for each sensor location, that is capable of diagnosing both gear and bearing faults.

3. Develop one generic FDN that can detect all problems, at all sensor locations.

The first approach was the least desirable of the three, and the third was the most desirable. The third approach seemed rather ambitious, and was likely to result in higher false alarm and false dismissal rates than the second approach. Based upon the results of the second approach, subsequent development could be pursued using the first or third. Thus we elected to synthesize new multi-case FDN's based on the second approach.

This first required creating a new set of five input data tables (one for each sensor location). These tables contained the same baseline data as used in the fault specific model synthesis, but included discrepant data from all five of the previously described seeded faults. After creation of these input tables, model synthesis followed the same strategy and procedures that were used for synthesis and test of the single fault FDN's.

The results of this process and the threshold optimization for each sensor's multi-case FDN are shown in Table VII. This shows that 3 out of 5 multi-case FDN's demonstrated false alarm rates of 1% or less when tested on data not used in training. If "3 out of 5" voting logic is applied to the thresholded output these five multi-case FDN's, the results should be sufficiently accurate.

Table VII: FDN Diagnostic Accuracy
(Case 1, 2, 3, 4, and 6)

Sensor Location	Evaluation Threshold	False Alarm %	False Dismissal %	Comments
1	0.95	1.00	14.00	
2	0.80	0.00	14.00	
3	0.50	2.30	0.00	
4	0.85	3.40	3.40	
5	0.50	0.00	3.00	most accurate multi-case FDN

Summary: The integration of SWAN with polynomial network modeling shows good potential for producing accurate diagnostics with low false alarm rates. The synthesized Fault Detection Networks (FDN's) are quite compact, and capable of being implemented by, small Distributed Processing Units (DPU's). These DPU's can function autonomously, or can be modular additions to existing aircraft systems such as Flight Data Recorders and Health Usage Monitoring Systems.

Ongoing work will include expansion of the training database to include additional seeded fault and baseline data. Additional testing will include evaluation of the best FDN's using baseline and fault data that not only was not in the training database, but was acquired from assembled gearboxes that were not in the training database. Frequency Domain Feature Extraction will be applied to the data, and used to train Fault Location and Fault Isolation Networks (FLN's & FIN's) that can pinpoint the root cause of an alarm from an FDN. Other networks for Percent Degradation, Regime Recognition and Sensor Validation are also being developed.